

(19)日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平9-319472

(43)公開日 平成9年(1997)12月12日

(51)Int.Cl. ⁶	識別記号	序内整理番号	F I	技術表示箇所
G 06 F 1/26			G 06 F 1/00	3 3 4 S
1/32			3/02	3 8 0 B
3/02	3 8 0		1/00	3 3 2 B
3/023			3/023	3 1 0 D
H 03 M 11/20				

審査請求 未請求 請求項の数1 FD (全5頁)

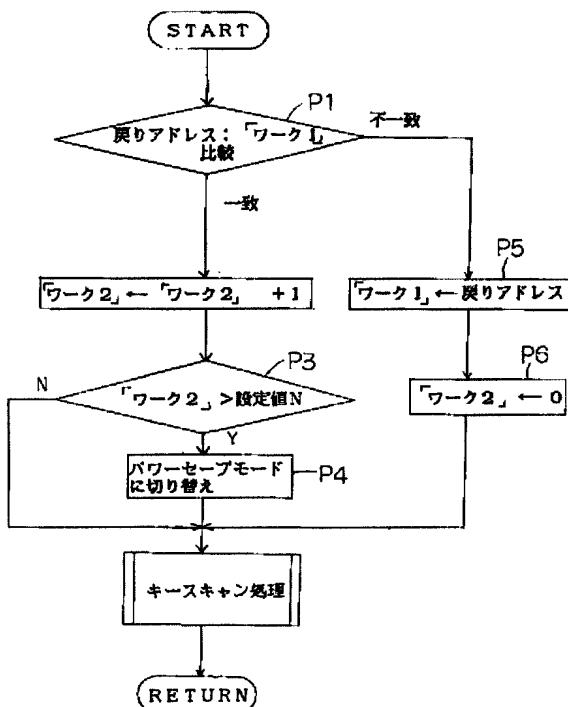
(21)出願番号 特願平8-160672	(71)出願人 日本電気ホームエレクトロニクス株式会社 大阪府大阪市中央区城見一丁目4番24号
(22)出願日 平成8年(1996)5月31日	(71)出願人 日本電気株式会社 東京都港区芝五丁目7番1号
	(72)発明者 吉村 成弘 大阪府大阪市中央区城見1丁目4番24号 日本電気ホームエレクトロニクス株式会社 内
	(74)代理人 弁理士 加川 征彦
	最終頁に続く

(54)【発明の名称】 情報処理装置

(57)【要約】

【課題】 アプリケーションプログラムの変更を必要とせずに、かつアプリケーションプログラムの性能を阻害せずに、パワーセーブ処理を実現する。

【解決手段】 キースキャンを発行したアプリケーションプログラムの位置(戻りアドレス)をRAMの「ワーク1」に記憶する。その後アプリケーションプログラムがサブルーチンを発行した時、その戻りアドレスと「ワーク1」のデータとを比較する。両者が一致した時はアプリケーションプログラムの同じ位置からの呼び出しであるから、連続してキースキャンを行っていることを意味する。そのキースキャン回数をカウントし、カウント数が設定値Xを超えた時、パワーセーブ処理を行う。外部ハードウェアからの割り込みによりパワーセーブを解除する。アプリケーションプログラムの戻りアドレスが同一かどうかで、連続したキースキャンかどうかを判断するので、システムプログラム側でパワーセーブ処理が可能である。



【特許請求の範囲】

【請求項 1】 アプリケーションプログラムがキースキャンルーチンをコールした時の当該アプリケーションプログラムの戻りアドレスを記憶装置に記憶し、その後にアプリケーションプログラムがサブルーチンをコールした時の当該アプリケーションプログラムの戻りアドレスと前記記憶装置のデータとを比較することにより、アプリケーションプログラムが連続してキースキャンルーチンをコールしているか否かを判断する連続キースキャン判断手段と、

前記連続キースキャン判断手段の判断結果に基づいて連続したキースキャンルーチンコールの回数をカウントし、そのカウント数が設定値を超えた時に、パワーセーブ信号を出すパワーセーブ指示手段と、

前記パワーセーブ信号によりパワーセーブモードに移行するとともに、外部ハードウェアからの割り込み信号によりパワーセーブモードを解除する電力制御回路とを備えたことを特徴とする情報処理装置。

【発明の詳細な説明】**【0001】**

【発明の属する技術分野】 本発明は情報処理装置に関し、特に、省電力システムが要求されるバッテリ動作の小型情報処理装置に適用して好適な情報処理装置に関する。

【0002】

【従来の技術】 従来、バッテリで動作する小型情報処理装置に搭載するプログラムは、バッテリ動作を前提としてパワーセーブ（省電力）を考慮した設計を行っている。すなわち、人の操作を待つような時にパワーセーブモードに切り替えるプログラムをアプリケーションプログラム自体に組み込んでおり、待ち時間中の消費電力を抑えてバッテリを長持ちさせるようにしている。一方、近年、アプリケーションプログラムの規模は大きくなる傾向にあるが、例えば日本語ワープロプログラムや表計算プログラムのような大規模アプリケーションプログラムの多くは、バッテリで動作する前提では設計されていない。すなわち、アプリケーションプログラム自体にパワーセーブ処理プログラムは組み込んではない。

【0003】

【発明が解決しようとする課題】 上述したバッテリ動作を前提としていない大規模アプリケーションプログラムをバッテリ動作の小型情報処理装置に搭載すると、パワーセーブ動作が行われないので、バッテリ寿命が短くなるという欠点があった。また、そのような大規模アプリケーションプログラムを改造してバッテリ動作の小型情報処理装置に対応させるのは、プログラムの規模が大きいため容易ではない。そのため、大規模アプリケーションプログラムを小型情報処理装置に搭載する場合のパワーセーブは、キー待ちを行う場合に限ってシステムプログラムの部分でパワーセーブモードに切り替えるという

手法が採用されることになる。しかし、アプリケーションプログラムはキースキャンを行いながら動作する部分が多いため、この方式はあまり適当でない。すなわち、キースキャンにおけるパワーセーブでは、取り消し（中止）キーをチェックしながら動作するプログラムが停止してしまったり、プログラムの動作が遅くなったりするという欠点がある。これを避けるためにはアプリケーションプログラムでパワーセーブを許可したり禁止したりする対応が必要になるが、前述のように大規模アプリケーションプログラムでは困難である。

【0004】 本発明は上記従来の欠点を解消するためになされたもので、アプリケーションプログラムの改造によらずに、かつアプリケーションプログラムの性能を損なうことなく、パワーセーブ処理を自動的にかつ適切に行うことができる情報処理装置を提供することにある。

【0005】

【課題を解決するための手段】 上記課題を解決する本発明の情報処理装置は、アプリケーションプログラムがキースキャンルーチンをコールした時の当該アプリケーションプログラムの戻りアドレスを記憶装置に記憶し、その後にアプリケーションプログラムがサブルーチンをコールした時の当該アプリケーションプログラムの戻りアドレスと前記記憶装置のデータとを比較することにより、アプリケーションプログラムが連続してキースキャンルーチンをコールしているか否かを判断する連続キースキャン判断手段と、前記連続キースキャン判断手段の判断結果に基づいて連続したキースキャンルーチンコールの回数をカウントし、そのカウント数が設定値を超えた時に、パワーセーブ信号を出すパワーセーブ指示手段と、前記パワーセーブ信号によりパワーセーブモードに移行するとともに、外部ハードウェアからの割り込み信号によりパワーセーブモードを解除する電力制御回路とを備えたことを特徴とする。

【0006】

【発明の実施の形態】 以下、本発明の実施の形態を図1～図4に示す一実施例を参照して説明する。図1に本発明の一実施例の小型情報処理装置のブロック図を示す。この小型情報処理装置は、中央処理装置（C P U）1と、周辺回路2と、電力制御回路3とを備えている。前記C P U 1および周辺回路2は、マイコンやパソコンで使用される一般的なものである。周辺回路2は、R O M 4、R A M 5を含み、システムプログラムおよびアプリケーションプログラムはR O M 4あるいはR A M 5に格納される。前記電力制御回路3はパワーセーブおよびパワーセーブモードの解除を行うものであり、C P U 1からのパワーセーブ信号に基づいてC P U 1へのクロックを停止（すなわちパワーセーブに移行）し、また外部ハードウェアからの割り込みがトリガとなって再起動（すなわちパワーセーブを解除）する回路を備えている。前記外部ハードウェア割り込みの要因には、例えば「キ

40 記C P U 1および周辺回路2は、マイコンやパソコンで使用される一般的なものである。周辺回路2は、R O M 4、R A M 5を含み、システムプログラムおよびアプリケーションプログラムはR O M 4あるいはR A M 5に格納される。前記電力制御回路3はパワーセーブおよびパワーセーブモードの解除を行うものであり、C P U 1からのパワーセーブ信号に基づいてC P U 1へのクロックを停止（すなわちパワーセーブに移行）し、また外部ハードウェアからの割り込みがトリガとなって再起動（すなわちパワーセーブを解除）する回路を備えている。前記外部ハードウェア割り込みの要因には、例えば「キ

3

一」、「タイマ」、「シリアルポート」等がある。

【0007】本発明の情報処理装置は、アプリケーションプログラムがキースキャンルーチンをコールした時の当該アプリケーションプログラムの戻りアドレスを記憶装置に記憶し、その後にアプリケーションプログラムがサブルーチンをコールした時の当該アプリケーションプログラムの戻りアドレスと前記記憶装置のデータとを比較することにより、アプリケーションプログラムが連続してキースキャンルーチンをコールしているか否かを判断する連続キースキャン判断手段と、前記連続キースキャン判断手段の判断結果に基づいて連続したキースキャンルーチンコールの回数をカウントし、そのカウント数が設定値を越えた時に、パワーセーブ信号を出すパワーセーブ指示手段と、前記パワーセーブ信号によりパワーセーブモードに移行するとともに、外部ハードウェアからの割り込み信号によりパワーセーブモードを解除する電力制御回路とを備えている。

【0008】上述の連続キースキャン判断手段において必要な戻りアドレスの検出手段について説明する。アプリケーションプログラムがキースキャンを行う場合、サブルーチンコールという手法を取るのが一般的である。サブルーチンコールはスタックメモリに戻りアドレスをセットしてキースキャンルーチンにジャンプする。この実施例では、前記サブルーチンコールの手法を利用する。この実施例において戻りアドレスを検出する際のRAM5にあるスタックメモリ（以下単にスタックという）の状況を図2（イ）または（ロ）に示す。また、図2（イ）または図2（ロ）のいずれも場合でもアプリケーションプログラムの戻りアドレスを正しく検出するための戻りアドレス検出処理のフローチャートを図3に示す。

【0009】図2（イ）は、アプリケーションプログラムがキースキャンルーチンとしてキーB IOS（キー処理における基本的な処理を受け持つプログラム）等の下位ルーチンのみを使用する場合のスタックの状況を示すものである。この場合、戻りアドレスは必ずスタックの先頭（5-A）にあるので発行元アドレスの検出は容易である。図2（ロ）は、アプリケーションプログラムがイベントマネージャ（イベント発生を管理するルーチン）等の上位ルーチンを使用し、イベントマネージャがキーB IOSを呼ぶ場合のスタックの状況を示すものである。この場合、スタックの先頭（5-A）にあるアドレスはイベントマネージャがキーB IOSを呼んでいるアドレスになっているので常に同じアドレスとなっている。したがって、発行元アドレス（アプリケーションプログラムの戻りアドレス）を検出するには、スタックの奥（5-B）にある戻りアドレスを見なければならぬ。そして、種々のアプリケーションプログラムに対応できるためには、上記の図2（イ）の状況および図2（ロ）の状況のいずれにも対応できる必要がある。イベ

4

ントマネージャはシステムプログラムの一種であるので、そのアドレスはアプリケーションプログラムとは関係なく、予め知ることができる。したがって、図2

（イ）の状況か図2（ロ）の状況かは、スタックの先頭（5-A）にあるアドレスがイベントマネージャのアドレスなのかどうかで判断することができる。したがって、図3のようなフローチャートで戻りアドレスの検出を行うとよい。すなわち、まず、スタックの先頭（5-A）のデータをポインタ（スタックポインタ）に格納する（ステップS1）。次いで、このポインタの内容を判断する（S2）。ポインタの内容（スタックの先頭（5-A）にあるアドレス）がイベントマネージャのアドレスと異なる場合（「その他のアドレス」の場合）、当該先頭（5-A）にあるアドレスを戻りアドレスと判断する。イベントマネージャのアドレスと同一の場合、スタックの奥（5-B）のアドレスを戻りアドレスと判断する（そして、スタック（5-B）にあるアドレスをポインタに格納する（S3））。

【0010】上述のパワーセーブ指示手段の一例を図4のフローチャートに示す。この実施例ではRAM5の「ワークエリア1」および「ワークエリア2」を利用している。アプリケーションプログラムがキースキャンルーチンをコールすると、その時の当該アプリケーションプログラムの戻りアドレス（図2（イ）のスタック（5-A）または図2（ロ）のスタック（5-B）にある戻りアドレス）と「ワークエリア（以下ワークという）1」のデータとを比較する（ステップP1）。

【0011】前記ステップP1においてスタックの戻りアドレスと「ワーク1」のデータの両者が一致した時は、「ワーク2」のデータをカウントアップ（+1）する（P2）。すなわち、1回としてカウントする。なお、「ワーク2」はカウンタとして用いている。スタックの戻りアドレスと「ワーク1」の値の両者が一致したことは、アプリケーションプログラムにおける同じ位置からの呼び出しであることを意味するので、「ワーク2」のデータ（すなわちカウント数）は、連続するキースキャン回数を意味し、キー待ち状態の長さに相当する。次いで、当該「ワーク2」の値を適切に定めたある設定値Nと比較する（P3）。「ワーク2」の値が設定値Nを越えていなければ、キースキャンが繰り返されて、前述の通り、連続のキースキャン回数がカウントされる。「ワーク2」の値が設定値Nを超えた時、パワーセーブ信号を電力制御回路3に出して、パワーセーブモードに切り替える（P4）。

【0012】そして、キー入力割り込み、タイマ入力割り込み、シリアル入力割り込み等が発生すると、電力制御回路3はその割り込み信号によりパワーセーブモードを解除する。

【0013】前記ステップP1において、スタックの戻りアドレスと「ワーク1」のデータの両者が一致しない

時は、アプリケーションプログラムにおける異なる位置からの呼び出しであることを意味するので、スタックの戻りアドレスを「ワーク1」に格納し（P5）、「ワーク2」の値を0に初期化する（P6）。なお、先の説明では、この初期化した状態から説明した。そして、その後パワーセーブは行わない通常処理を行う。

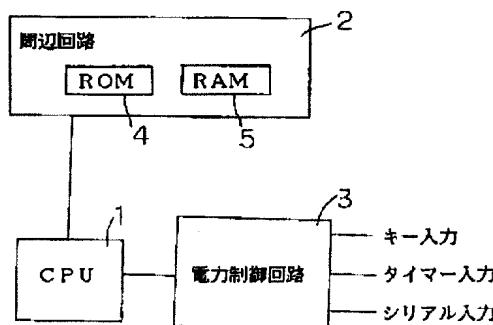
【0014】なお、パワーセーブモードに切り替える設定値Nは、適切なパワーセーブ動作を行うために重要であるが、この設定値Nは、取り消し（中止）キーをチェックしながら動作するルーチンのキースキャン回数の最大値を調べることで適切に設定できる。また、パワーセーブモードに切り替える設定値Nがアプリケーションプログラムが処理するデータに依存する場合、あるいは、非常に大きい場合は、パワーセーブモードをオン・オフの2モードではなく、例えばCPUクロック周波数を10通りに変えられるようなハードウェアにしておき、上記「ワーク2」の値に応じて、段階的にクロック周波数を下げるという構成にしてもよい。

【0015】なお、アプリケーションプログラムがキースキャンルーチンをコールした時の当該アプリケーションプログラムの位置（戻りアドレス）を検出する連続キースキャン判断手段は、上述の実施例の方法に限定されない。

【0016】

【発明の効果】以上説明したように、本発明によれば、キースキャンを発行したアプリケーションプログラムの戻りアドレスが同一かどうかで、連続してキースキャンを行っているか否かを判断し、その連続のキースキャン回数に基づいて、パワーセーブ動作をする構成であるので、アプリケーションプログラムの改造を必要とせず、システムプログラム側でパワーセーブ処理を実現でき、かつ、バッテリ動作を前提としない大規模アプリケーシ

【図1】



ョンを小型情報処理装置に搭載した場合でも、バッテリを長持ちさせることができるといった効果がある。

【0017】また、戻りアドレスを利用する本発明の構成は、極めて単純であり、小型情報処理装置に実際に適用することが極めて容易であり、実用価値が高いという効果がある。

【図面の簡単な説明】

【図1】本発明の一実施例を示す小型情報処理装置のブロック構成図である。

10 【図2】(イ)および(ロ)はいずれもアプリケーションプログラムのアドレスが格納されるスタックのイメージを示す図であり、(イ)はアプリケーションプログラムがキーボードB IOSなどの下位ルーチンのみを使用する場合のもの、(ロ)はアプリケーションプログラムがイベントマネージャ等の上位ルーチンを使用し、このイベントマネージャがキーB IOSを呼ぶ場合のものである。

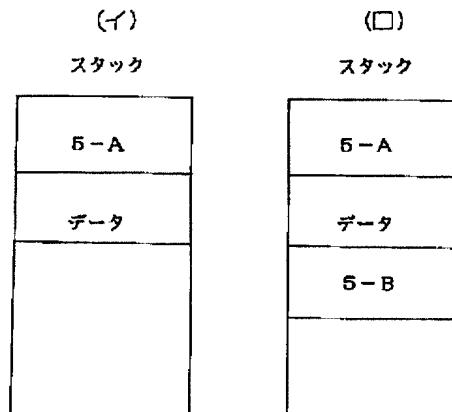
【図3】本発明の実施例において、アプリケーションプログラムがシステムプログラムのサブルーチンを呼び出した時に、システムプログラムがアプリケーションプログラムの戻りアドレスを検出する戻りアドレス検出処理のフローチャートである。

20 【図4】上記実施例の小型情報処理装置のシステムプログラムにおけるキースキャンルーチンに伴う処理を示すフローチャートである。

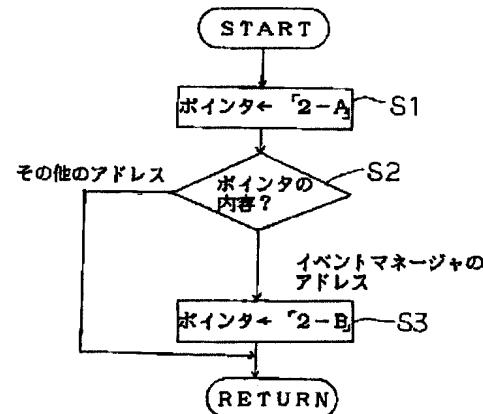
【符号の説明】

1	中央処理装置 (CPU)
2	周辺機器
3	電力制御回路
30	RAM
5	ROM

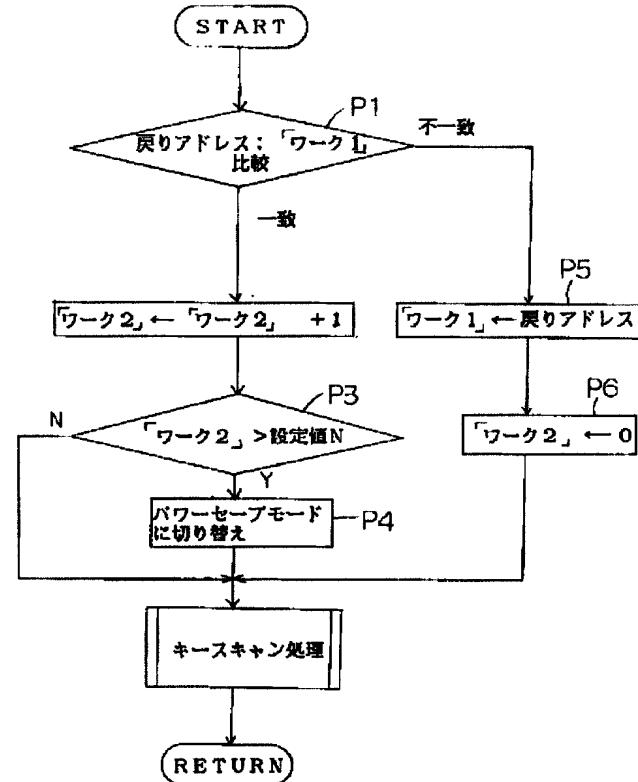
【図2】



【図3】



【図4】



フロントページの続き

(72)発明者 増野 博志

東京都港区芝5丁目7番1号 日本電気株
式会社内

(72)発明者 池田 敏治

大阪府大阪市中央区城見1丁目4番24号
日本電気ホームエレクトロニクス株式会社
内